DroMOOC

Trajectory planning

Basic Level

Bézier and B-Spline curves: Introduction

Gauthier ROUSSEAU

L2S/CentraleSupélec

Introduction

Goal of this presentation

- Present one way to parameterize **polynomial** and **piecewise polynomial trajectories**
- **Overview** of Bézier and B-splines curves: present **some** of the main properties of these curves

Bézier curves

Bézier curves

Introduction

Introduced in the sixties for CAD

- 1958 Paul de Casteljau (ENS), engineer at Citröen
- 1962 Pierre Bézier (Arts&Métiers/Supélec), engineer at Renault

Intuitive way to parameterize polynomials - focused on the **shape** of the curve

Bézier curves - Definition

Definition

Let be a set of $(n+1)$ **control points** $\mathbf{P} = \{\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_n\}$, with $n \in \mathbb{N}$ and $\mathbf{p}_i \in \mathbb{R}^d$
This set defines the **Bézier curve**

$$\mathcal{B}_{\mathbf{P}} : \left(\begin{array}{ccc} \mathbb{R} & \rightarrow & \mathbb{R}^d \\ \tau & \mapsto & \displaystyle\sum_{i=0}^{n} b_{i,n}(\tau)\ \mathbf{p}_i \end{array} \right) \tag{1}$$

with $b_{i,n}(\tau) = \binom{n}{i}\ \tau^i (1-\tau)^{n-i} = \frac{n!}{i!\,(n-i)!}\ \tau^i (1-\tau)^{n-i}$

- The curve is a polynomial of degree n
- The $b_{i,n}$ are the **Bernstein** polynomials of degree n
- $\mathcal{B}_{\mathbf{P}}(0) = \mathbf{p}_0$ and $\mathcal{B}_{\mathbf{P}}(1) = \mathbf{p}_n$

Bézier curves - Definition

Definition

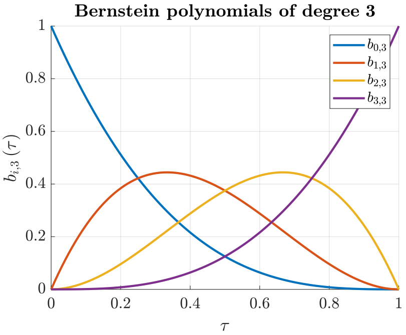
Let be a set of $(n+1)$ **control points** $\mathbf{P} = \{\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_n\}$, with $n \in \mathbb{N}$ and $\mathbf{p}_i \in \mathbb{R}^d$
This set defines the **Bézier curve**

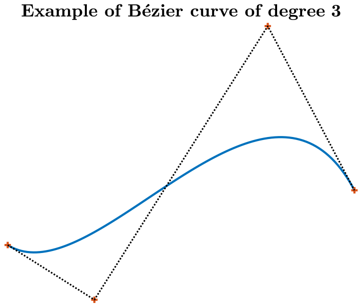
$$\boldsymbol{\mathcal{B}}_{\mathbf{P}} : \left(\begin{array}{ccc} [0,1] & \rightarrow & \mathbb{R}^d \\ \tau & \mapsto & \displaystyle\sum_{i=0}^{n} b_{i,n}(\tau)\ \mathbf{p}_i \end{array} \right) \tag{2}$$

with $b_{i,n}(\tau) = \binom{n}{i}\ \tau^i (1-\tau)^{n-i} = \frac{n!}{i!\,(n-i)!}\ \tau^i (1-\tau)^{n-i}$

- The curve is a polynomial of degree n
- The $b_{i,n}$ are the **Bernstein** polynomials of degree n
- $\boldsymbol{\mathcal{B}}_{\mathbf{P}}(0) = \mathbf{p}_0$ and $\boldsymbol{\mathcal{B}}_{\mathbf{P}}(1) = \mathbf{p}_n$

Bézier curves - Example





Example of Bézier curve of control points

$$\mathbf{P} = \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \begin{pmatrix} 3 \\ 4 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \end{pmatrix} \right\} \text{ (degree 3)}$$

Bézier curves - Affine parameter transformation

An affine parameter transformation allows to change the interval of definition

Example

Parameterizing a trajectory $\zeta : [t_0, t_1] \to \mathbb{R}^d$ with the Bézier curve $\boldsymbol{\mathcal{B}}_{\mathbf{P}}$
Let be

$$\tau : \begin{pmatrix} [t_0, t_1] \to & [0, 1] \\ t \mapsto & \dfrac{t - t_0}{t_1 - t_0} \end{pmatrix} \tag{3}$$

Then $\forall t \in [t_0, t_1] \;\; \zeta(t) = \boldsymbol{\mathcal{B}}_{\mathbf{P}}(\tau(t))$

Derivatives/Integrals scaling

- $\frac{\mathrm{d}^l}{\mathrm{d}t^l}\zeta(t) = \frac{1}{(t_1 - t_0)^l} \frac{\mathrm{d}^l}{\mathrm{d}\tau^l} \boldsymbol{\mathcal{B}}_{\mathbf{P}}(\tau(t))$
- $\int_a^b \zeta(u)\,\mathrm{d}u = (t_1 - t_0) \int_{\tau(a)}^{\tau(b)} \boldsymbol{\mathcal{B}}_{\mathbf{P}}(v)\,\mathrm{d}v$

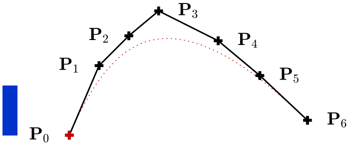
Bézier curves - Barycenter interpretation

Bernstein poylnomials verify

$$\forall \tau \in [0,1] : \begin{cases} b_{i,n}(\tau) \geqslant 0 \\ \displaystyle\sum_{i=0}^{n} b_{i,n}(\tau) = 1 \end{cases} \tag{4}$$

Barycenter interpretation

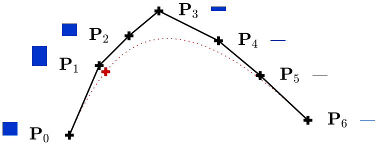
A Bézier curve of degree n is a **convex combination** of $(n+1)$ control points, weighted by **Bernstein polynomials** of degree n



Bézier curves - Barycenter interpretation

Bernstein poylnomials verify

$$\forall \tau \in [0,1] : \begin{cases} b_{i,n}(\tau) \geqslant 0 \\ \sum_{i=0}^{n} b_{i,n}(\tau) = 1 \end{cases} \qquad (5)$$



Barycenter interpretation

A Bézier curve of degree n is a **convex combination** of $(n+1)$ control points, weighted by **Bernstein polynomials** of degree n

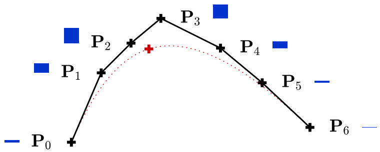
Bézier curves - Barycenter interpretation

Bernstein poylnomials verify

$$\forall \tau \in [0,1] : \begin{cases} b_{i,n}(\tau) \geqslant 0 \\ \sum_{i=0}^{n} b_{i,n}(\tau) = 1 \end{cases} \tag{6}$$

Barycenter interpretation

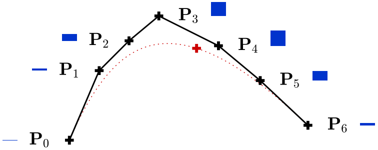
A Bézier curve of degree n is a **convex combination** of $(n+1)$ control points, weighted by **Bernstein polynomials** of degree n



Bézier curves - Barycenter interpretation

Bernstein poylnomials verify

$$\forall \tau \in [0,1] : \begin{cases} b_{i,n}(\tau) \geqslant 0 \\ \sum_{i=0}^{n} b_{i,n}(\tau) = 1 \end{cases} \tag{7}$$



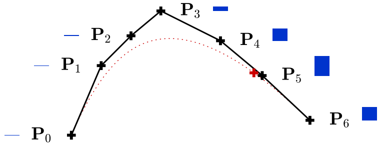
Barycenter interpretation

A Bézier curve of degree n is a **convex combination** of $(n+1)$ control points, weighted by **Bernstein polynomials** of degree n

Bézier curves - Barycenter interpretation

Bernstein poylnomials verify

$$\forall \tau \in [0,1] : \begin{cases} b_{i,n}(\tau) \geqslant 0 \\ \sum_{i=0}^{n} b_{i,n}(\tau) = 1 \end{cases} \tag{8}$$



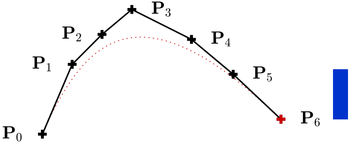
Barycenter interpretation

A Bézier curve of degree n is a **convex combination** of $(n+1)$ control points, weighted by **Bernstein polynomials** of degree n

Bézier curves - Barycenter interpretation

Bernstein poylnomials verify

$$\forall \tau \in [0,1] : \begin{cases} b_{i,n}(\tau) \geqslant 0 \\ \sum_{i=0}^{n} b_{i,n}(\tau) = 1 \end{cases} \tag{9}$$



Barycenter interpretation

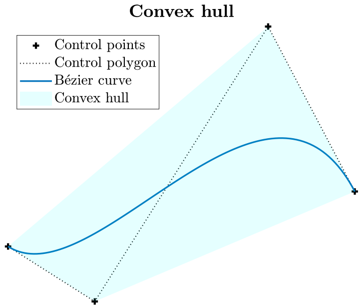
A Bézier curve of degree n is a **convex combination** of $(n+1)$ control points, weighted by **Bernstein polynomials** of degree n

Bézier curves - Convex hull property

Convex hull property

A Bézier curve **lies within the convex hull of its control points**

$$\forall \tau \in [0,1] \;\; \boldsymbol{\mathcal{B}}_{\mathbf{P}}(\tau) \in \mathrm{Conv}(\mathbf{P}) \tag{10}$$



- This is a direct consequence of (9)
- Can be used to constrain a curve into a convex region (!! only a **sufficient condition**, can induce conservatism)

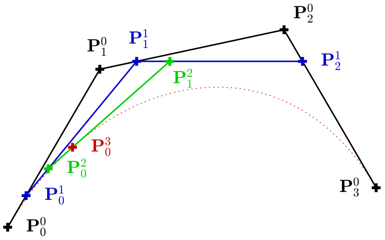
Bézier curves - De Casteljau algorithm

De Castetjau algorithm

Numerically stable evaluation of bézier curve by the mean of successive convex combinations

$$\mathcal{B}_{\mathbf{P}}(\tau) = \sum_{i=0}^{n} b_{i,n}(\tau)\ \mathbf{p}_i^0 = \sum_{i=0}^{n-1} b_{i,n-1}(\tau)\ \mathbf{p}_i^1 = \cdots = \sum_{i=0}^{0} b_{i,0}(\tau)\ \mathbf{p}_i^n = \mathbf{p}_i^n$$

with $\mathbf{p}_i^0 = \mathbf{p}_i$ and $\mathbf{p}_i^j = (1-\tau)\,\mathbf{p}_i^{j-1} + \tau\,\mathbf{p}_{i+1}^{j-1}$



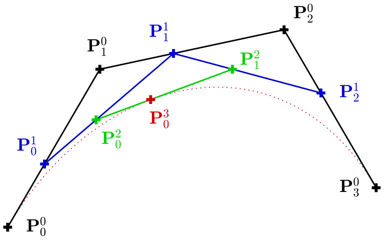
Bézier curves - De Casteljau algorithm

De Castetjau algorithm

Numerically stable evaluation of bézier curve by the mean of successive convex combinations

$$\mathcal{B}_{\mathbf{P}}(\tau) = \sum_{i=0}^{n} b_{i,n}(\tau)\ \mathbf{p}_i^0 = \sum_{i=0}^{n-1} b_{i,n-1}(\tau)\ \mathbf{p}_i^1 = \cdots = \sum_{i=0}^{0} b_{i,0}(\tau)\ \mathbf{p}_i^n = \mathbf{p}_i^n$$

with $\mathbf{p}_i^0 = \mathbf{p}_i$ and $\mathbf{p}_i^j = (1-\tau)\,\mathbf{p}_i^{j-1} + \tau\,\mathbf{p}_{i+1}^{j-1}$



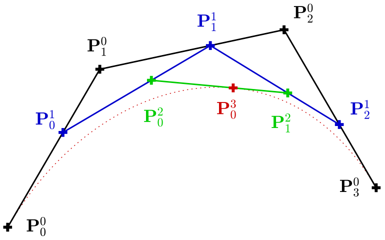
Bézier curves - De Casteljau algorithm

De Castetjau algorithm

Numerically stable evaluation of bézier curve by the mean of successive convex combinations

$$\mathcal{B}_{\mathbf{P}}(\tau) = \sum_{i=0}^{n} b_{i,n}(\tau)\ \mathbf{p}_i^0 = \sum_{i=0}^{n-1} b_{i,n-1}(\tau)\ \mathbf{p}_i^1 = \cdots = \sum_{i=0}^{0} b_{i,0}(\tau)\ \mathbf{p}_i^n = \mathbf{p}_i^n$$

with $\mathbf{p}_i^0 = \mathbf{p}_i$ and $\mathbf{p}_i^j = (1-\tau)\,\mathbf{p}_i^{j-1} + \tau\,\mathbf{p}_{i+1}^{j-1}$



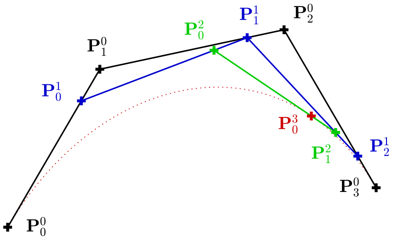
Bézier curves - De Casteljau algorithm

De Castetjau algorithm

Numerically stable evaluation of bézier curve by the mean of successive convex combinations

$$\mathcal{B}_{\mathbf{P}}(\tau) = \sum_{i=0}^{n} b_{i,n}(\tau)\ \mathbf{p}_i^0 = \sum_{i=0}^{n-1} b_{i,n-1}(\tau)\ \mathbf{p}_i^1 = \cdots = \sum_{i=0}^{0} b_{i,0}(\tau)\ \mathbf{p}_i^n = \mathbf{p}_i^n$$

with $\mathbf{p}_i^0 = \mathbf{p}_i$ and $\mathbf{p}_i^j = (1-\tau)\,\mathbf{p}_i^{j-1} + \tau\,\mathbf{p}_{i+1}^{j-1}$



Bézier curves - Polynomial interpretation

Bernstein polynomials of degree n are a basis of the vector space of the polynomials of degree n and less

Polynomial interpretation

A Bézier curve of degree n is a **polynomial** of degree n, defined on $[0,1]$ and **expressed in the basis of Bernstein polynomials** of degree n

Expression in monomial basis

$$\forall \tau \in [0,1] \;\; \boldsymbol{\mathcal{B}}_{\mathbf{P}}(\tau) = \sum_{i=0}^{n} \mathbf{c}_i \, \tau^i$$

with

$$\forall i \in [\![0,n]\!] \begin{cases} \mathbf{c}_i = \binom{n}{i} \sum_{j=0}^{i} (-1)^{i-j} \binom{i}{j} \mathbf{p}_j \\ \mathbf{p}_i = \binom{n}{i}^{-1} \sum_{j=0}^{i} \binom{n-j}{i-j} \mathbf{c}_i \end{cases}$$

Bézier curves - Derivative/Integral

Derivative

The derivative of a Bézier curve of degree $n > 0$ **is a also a Bézier curve**, of degree $n-1$ and whose control points are a linear combination of the original ones

$$\mathcal{B}'_{\mathbf{P}} = \mathcal{B}_{\mathbf{P}^{(1)}} \quad \text{with} \quad \forall i \in [\![0, n-1]\!] \;\; \mathbf{p}_i^{(1)} = n\left(\mathbf{p}_{i+1} - \mathbf{p}_i\right) \tag{11}$$

Integral

The integral of a Bézier curve of degree n **is a also a Bézier curve**, of degree $n+1$

Given an initial condition $\mathbf{p}_0^{(-1)}$

$$\int \mathcal{B}_{\mathbf{P}} = \mathcal{B}_{\mathbf{P}^{(-1)}} \quad \text{with} \quad \forall i \in [\![0, n]\!] \;\; \mathbf{p}_{i+1}^{(-1)} = \frac{1}{n+1}\mathbf{p}_i + \mathbf{p}_i^{(-1)} \tag{12}$$

Bézier curves - A powerful tool

- **Convex hull property**
- **Derivatives are Bézier curves whose control points are given by the original ones**

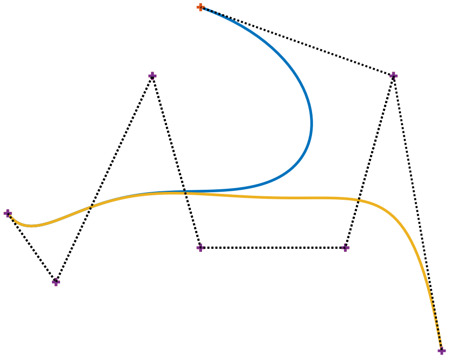
$\Rightarrow$ Powerful tool for constraining a trajectory and its derivatives/integrals in convex regions

Example

- Flight corridors
- Obstacle free convex regions
- Speed, acceleration limitation

B-splines curves

B-splines curves - Introduction: Piecewise polynomial curves



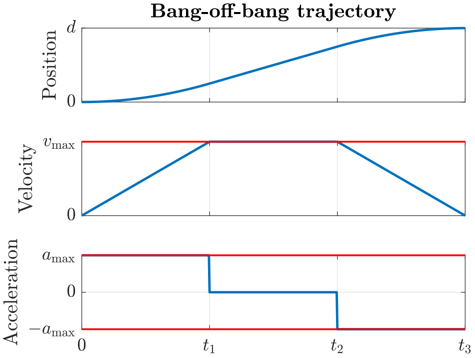
Bézier curves: Each control points impact the whole curve

B-splines curves - Introduction: Piecewise polynomial curves

Example

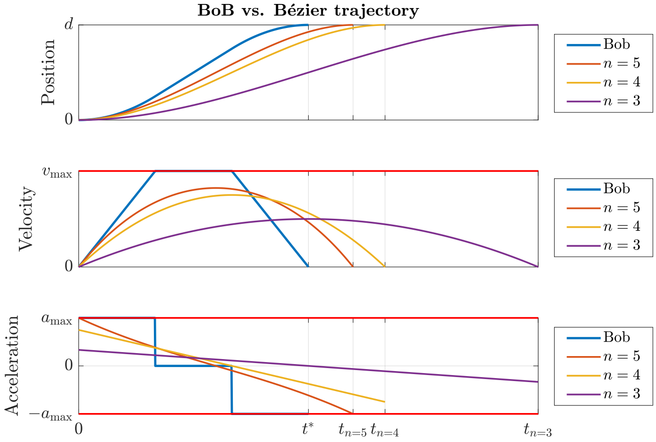
Generate a rest-to-rest trajectory to cover a given distance d in a minimum-time, with speed and acceleration limitations $v_{\max}$ and $a_{\max}$

Intuitive and optimal solution: Bang-off-bang acceleration law

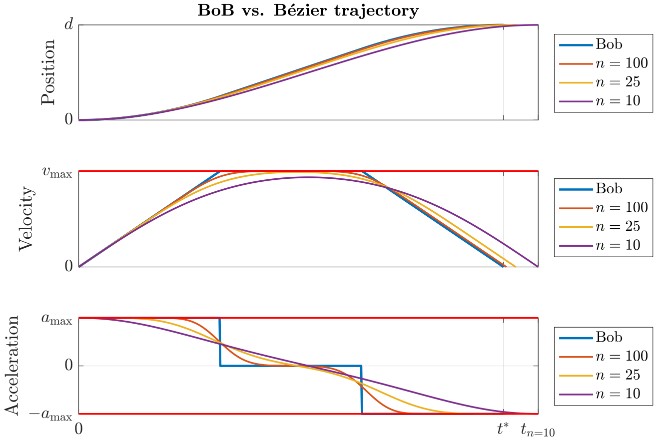


Simple piecewise degree 2 polynomial

B-splines curves - Introduction: Piecewise polynomial curves



B-splines curves - Introduction: Piecewise polynomial curves



Piecewise trajectory = more degrees of freedom

B-splines curves - Introduction

Introduction

- Introduced in the forties (Isaac J. Schoenberg)
- Developed in the seventies for CAD and computer graphics (Carl R. de Boor, Maurice Cox, Richard Riesenfeld, Wolfgang Boehm)
- Extension of Bézier curves to piecewise polynomials

B-splines curves - Introduction

Bézier curves

- Degree given by number of control points
- Defined on $[0, 1]$, need to use affine parametric transformation
- Each control point impacts the shape of the whole curve

B-spline curves

- Degree can be inferior to the number of control points
- Defined on arbitrary interval
- Piecewise, with adjustable continuity at the connections
- Local influence of control points

B-splines curves - Definition

Definition

- A set of $(n+1)$ **control points** $\mathbf{P} = \{\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_n\}$, with $n \in \mathbb{N}$ and $\mathbf{p}_i \in \mathbb{R}^d$
- A **polynomial degree** $k \in \mathbb{N}$, such that $k \leqslant n$
- A vector of $(m+1) = n+k+2$ **increasing knots** $\boldsymbol{\tau} = \begin{pmatrix} \tau_0 & \tau_1 & \ldots & \tau_m \end{pmatrix}$, with $\tau_i \in \mathbb{R}$

Define a B-spline curve

$$\boldsymbol{\mathcal{B}}_{\mathbf{P},\boldsymbol{\tau}} : \left(\begin{array}{cc} \mathbb{R} \to & \mathbb{R}^d \\ t \mapsto & \displaystyle\sum_{i=0}^{n} N_{i,k}^{\boldsymbol{\tau}}(t)\ \mathbf{p}_i \end{array} \right) \tag{13}$$

- The $N_{i,k}^{\boldsymbol{\tau}}$ are the Basis-splines (or B-splines) functions of degree k and knots $\boldsymbol{\tau}$

B-splines curves - Definition

Definition

- A set of $(n+1)$ **control points** $\mathbf{P} = \{\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_n\}$, with $n \in \mathbb{N}$ and $\mathbf{p}_i \in \mathbb{R}^d$
- A **polynomial degree** $k \in \mathbb{N}$, such that $k \leqslant n$
- A vector of $(m+1) = n+k+2$ **increasing knots** $\boldsymbol{\tau} = \begin{pmatrix} \tau_0 & \tau_1 & \ldots & \tau_m \end{pmatrix}$, with $\tau_i \in \mathbb{R}$

Define a B-spline curve

$$\boldsymbol{\mathcal{B}}_{\mathbf{P},\boldsymbol{\tau}} : \left(\begin{array}{rl} \color{red}{[\tau_k, \tau_{n+1}[} \rightarrow & \mathbb{R}^d \\ t \mapsto & \displaystyle\sum_{i=0}^{n} N_{i,k}^{\boldsymbol{\tau}}(t)\ \mathbf{p}_i \end{array} \right) \tag{14}$$

- The $N_{i,k}^{\boldsymbol{\tau}}$ are the Basis-splines (or B-splines) functions of degree k and knots $\boldsymbol{\tau}$

B-splines curves - Basis-splines functions

Definition

$\forall i \in [\![0, n]\!] \ \ \forall t \in \mathbb{R}$

If $k = 0$

$$N^{\boldsymbol{\tau}}_{i,k}(t) = \begin{cases} 1 & \text{if } \tau_i \leqslant t < \tau_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

Else

$$N^{\boldsymbol{\tau}}_{i,k}(t) = \omega^{\boldsymbol{\tau}}_{i,k}(t)\, N^{\boldsymbol{\tau}}_{i,k-1}(t) + \left(1 - \omega^{\boldsymbol{\tau}}_{i+1,k}(t)\right) N^{\boldsymbol{\tau}}_{i+1,k-1}(t)$$

with

$$\omega^{\boldsymbol{\tau}}_{i,k}(t) = \begin{cases} \dfrac{t - \tau_i}{\tau_{i+k} - \tau_i} & \text{if } \tau_{i+k} > \tau_i \\ 0 & \text{otherwise} \end{cases}$$

Support

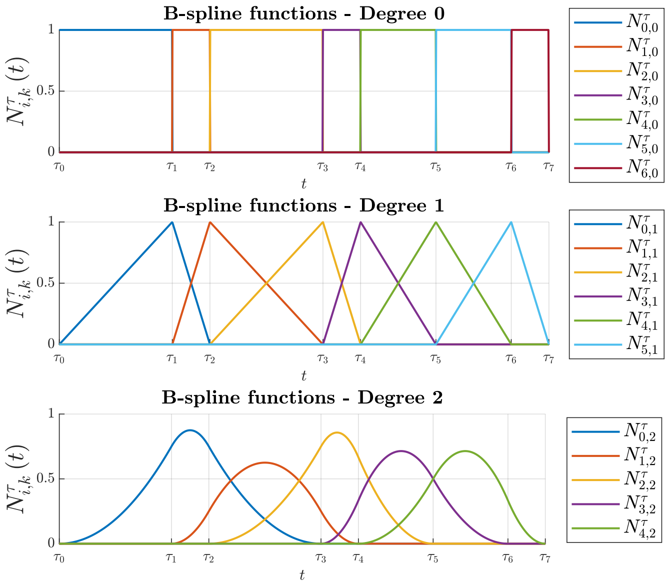
$N^{\boldsymbol{\tau}}_{i,k} = 0$ outside of $[\tau_i, \tau_{i+k+1}[$

Special case

If $\boldsymbol{\tau} = (\underbrace{0, \ldots, 0}_{k+1 \text{ knots}}, \underbrace{1, \ldots, 1}_{k+1 \text{ knots}})$

then $N^{\boldsymbol{\tau}}_{i,k} = b_{i,k}$

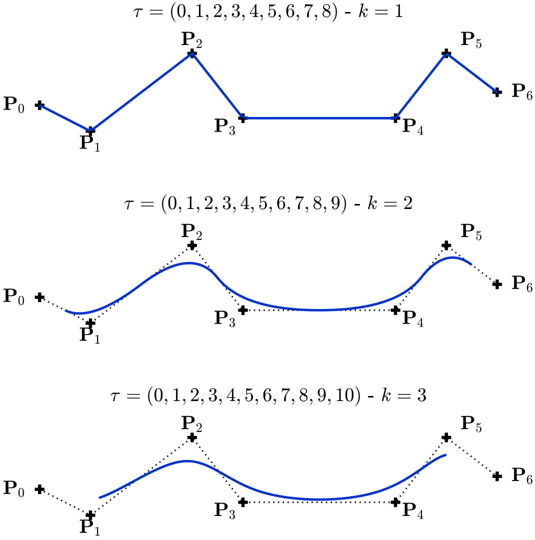
B-splines curves - Example



Example of B-spline basis functions with

$$\boldsymbol{\tau} = \begin{pmatrix} 0 & 3 & 4 & 7 & 8 & 10 & 12 & 13 \end{pmatrix}$$

B-splines curves - Example

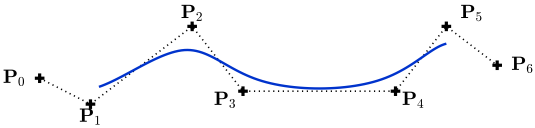


Example of B-spline curve with

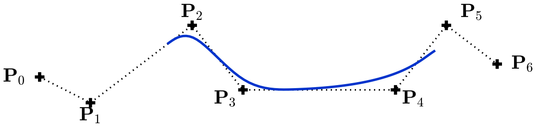
$$\mathbf{P} = \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 3 \\ 3 \end{pmatrix}, \begin{pmatrix} 4 \\ 0.5 \end{pmatrix}, \begin{pmatrix} 7 \\ 0.5 \end{pmatrix}, \begin{pmatrix} 8 \\ 3 \end{pmatrix}, \begin{pmatrix} 9 \\ 1.5 \end{pmatrix} \right\}$$

B-splines curves - Example

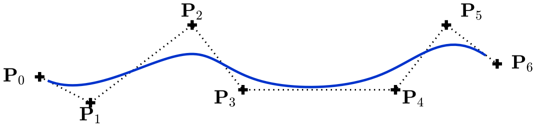
$\tau = (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10)$ - $k = 3$



$\tau = (0, 10, 20, 30, 31, 32, 33, 40, 50, 60, 70)$ - $k = 3$



$\tau = (0, 1, 2, 3, 20, 30, 40, 51, 52, 53, 54)$ - $k = 3$



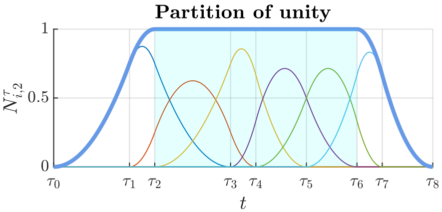
Example of B-spline curve with

$$\mathbf{P} = \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 3 \\ 3 \end{pmatrix}, \begin{pmatrix} 4 \\ 0.5 \end{pmatrix}, \begin{pmatrix} 7 \\ 0.5 \end{pmatrix}, \begin{pmatrix} 8 \\ 3 \end{pmatrix}, \begin{pmatrix} 9 \\ 1.5 \end{pmatrix} \right\}$$

B-splines curves - Convex hull property

B-splines verify

$$\begin{cases} \forall i \in [0, n] \quad N^{\tau}_{i,k} \geqslant 0 \\ \forall j \in [\![k, n]\!] \quad \forall t \in [\tau_j, \tau_{j+1}[\quad \displaystyle\sum_{i=0}^{n} N^{\tau}_{i,k}(t) = \sum_{i=j}^{j+k+1} N^{\tau}_{i,k}(t) = 1 \end{cases} \tag{15}$$



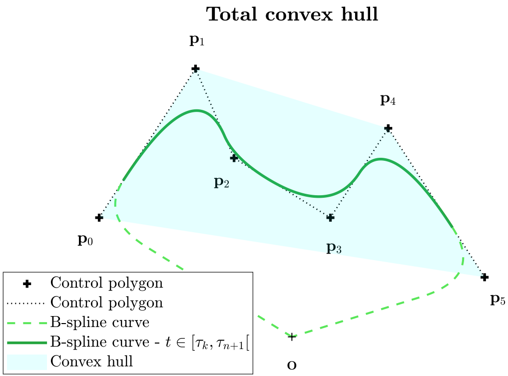
Convex hull property

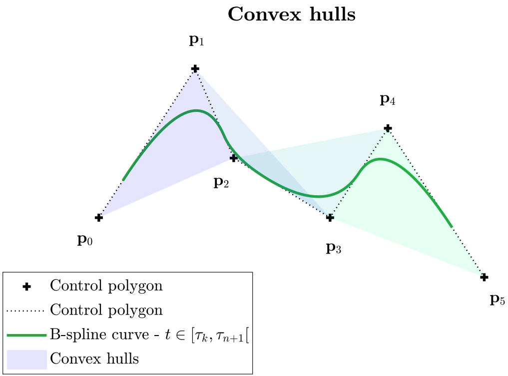
$\forall j \in [\![k, n]\!] \quad \forall t \in [\tau_j, \tau_{j+1}[$

$$\mathcal{B}_{\mathbf{P},\boldsymbol{\tau}}(t) \in \mathrm{Conv}\big(\{\mathbf{p}_i \mid i \in [\![j-k, j]\!]\} \big) \tag{16}$$

$$\Big(\underbrace{\tau_0, \dots, \tau_{k-1},}_{k \text{ knots}} \underset{\substack{\downarrow \\ \text{starting} \\ \text{knot}}}{\tau_k}, \underbrace{\tau_{k+1} \dots, \tau_n,}_{\substack{n-k \\ \text{internal knots}}} \underset{\substack{\downarrow \\ \text{ending} \\ \text{knot}}}{\tau_{n+1}}, \underbrace{\tau_{n+2}, \dots, \tau_m}_{k \text{ knots}}\Big)$$

B-splines curves - Example





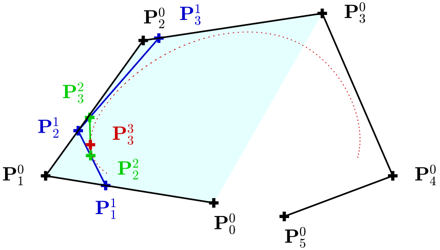
B-splines curves - De Boor - Cox algorithm

De Boor - Cox algorithm

Extension of de Casteljau algorithm to B-splines: $\forall j \in [\![k, n]\!]\ \ \forall t \in [\tau_j, \tau_{j+1}[$

$$\mathcal{B}_{\mathbf{P},\boldsymbol{\tau}}(t) = \sum_{i=j-k}^{j} N_{i,k}^{\boldsymbol{\tau}}(t)\ \mathbf{p}_i^0 = \sum_{i=j-k+1}^{j} N_{i,k-1}^{\boldsymbol{\tau}}(t)\ \mathbf{p}_i^1 = \cdots = \sum_{i=j}^{j} N_{i,0}^{\boldsymbol{\tau}}(t)\ \mathbf{p}_i^0 k = \mathbf{p}_j^0 k$$

with $\mathbf{p}_i^0 = \mathbf{p}_i$ and $\mathbf{p}_i^l = \omega_{i,k-l}^{\boldsymbol{\tau}}(t)\, \mathbf{p}_i^{l-1} + \left(1 - \omega_{i,k-l}^{\boldsymbol{\tau}}(t)\right) \mathbf{p}_{i-1}^{l-1}$



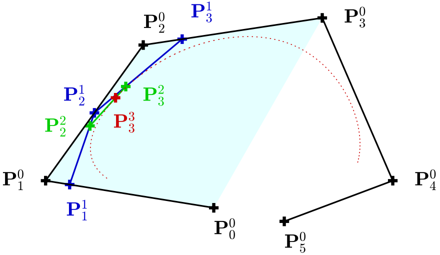
B-splines curves - De Boor - Cox algorithm

De Boor - Cox algorithm

Extension of de Casteljau algorithm to B-splines: $\forall j \in [\![k, n]\!] \;\; \forall t \in [\tau_j, \tau_{j+1}[$

$$\mathcal{B}_{\mathbf{P},\boldsymbol{\tau}}(t) = \sum_{i=j-k}^{j} N^{\boldsymbol{\tau}}_{i,k}(t)\ \mathbf{p}_i^0 = \sum_{i=j-k+1}^{j} N^{\boldsymbol{\tau}}_{i,k-1}(t)\ \mathbf{p}_i^1 = \cdots = \sum_{i=j}^{j} N^{\boldsymbol{\tau}}_{i,0}(t)\ \mathbf{p}_i^0 k = \mathbf{p}_j^0 k$$

with $\mathbf{p}_i^0 = \mathbf{p}_i$ and $\mathbf{p}_i^l = \omega^{\boldsymbol{\tau}}_{i,k-l}(t)\, \mathbf{p}_i^{l-1} + \left(1 - \omega^{\boldsymbol{\tau}}_{i,k-l}(t)\right) \mathbf{p}_{i-1}^{l-1}$



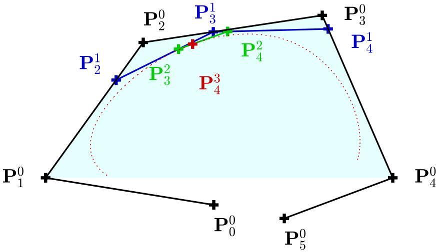
B-splines curves - De Boor - Cox algorithm

De Boor - Cox algorithm

Extension of de Casteljau algorithm to B-splines: $\forall j \in [\![k, n]\!] \;\; \forall t \in [\tau_j, \tau_{j+1}[$

$$\mathcal{B}_{\mathbf{P},\boldsymbol{\tau}}(t) = \sum_{i=j-k}^{j} N^{\boldsymbol{\tau}}_{i,k}(t)\ \mathbf{p}_i^0 = \sum_{i=j-k+1}^{j} N^{\boldsymbol{\tau}}_{i,k-1}(t)\ \mathbf{p}_i^1 = \cdots = \sum_{i=j}^{j} N^{\boldsymbol{\tau}}_{i,0}(t)\ \mathbf{p}_i^0 k = \mathbf{p}_j^0 k$$

with $\mathbf{p}_i^0 = \mathbf{p}_i$ and $\mathbf{p}_i^l = \omega^{\boldsymbol{\tau}}_{i,k-l}(t)\,\mathbf{p}_i^{l-1} + \left(1 - \omega^{\boldsymbol{\tau}}_{i,k-l}(t)\right)\mathbf{p}_{i-1}^{l-1}$



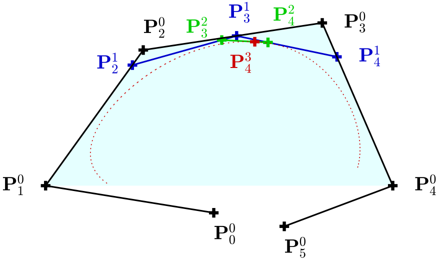
B-splines curves - De Boor - Cox algorithm

De Boor - Cox algorithm

Extension of de Casteljau algorithm to B-splines: $\forall j \in [\![k, n]\!] \ \ \forall t \in [\tau_j, \tau_{j+1}[$

$$\mathcal{B}_{\mathbf{P},\boldsymbol{\tau}}(t) = \sum_{i=j-k}^{j} N^{\boldsymbol{\tau}}_{i,k}(t)\ \mathbf{p}^0_i = \sum_{i=j-k+1}^{j} N^{\boldsymbol{\tau}}_{i,k-1}(t)\ \mathbf{p}^1_i = \cdots = \sum_{i=j}^{j} N^{\boldsymbol{\tau}}_{i,0}(t)\ \mathbf{p}^0_i k = \mathbf{p}^0_j k$$

with $\mathbf{p}^0_i = \mathbf{p}_i$ and $\mathbf{p}^l_i = \omega^{\boldsymbol{\tau}}_{i,k-l}(t)\, \mathbf{p}^{l-1}_i + \left(1 - \omega^{\boldsymbol{\tau}}_{i,k-l}(t)\right) \mathbf{p}^{l-1}_{i-1}$



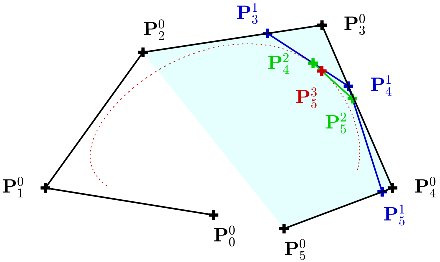
B-splines curves - De Boor - Cox algorithm

De Boor - Cox algorithm

Extension of de Casteljau algorithm to B-splines: $\forall j \in [\![k, n]\!] \;\; \forall t \in [\tau_j, \tau_{j+1}[$

$$\mathcal{B}_{\mathbf{P},\boldsymbol{\tau}}(t) = \sum_{i=j-k}^{j} N_{i,k}^{\boldsymbol{\tau}}(t)\ \mathbf{p}_i^0 = \sum_{i=j-k+1}^{j} N_{i,k-1}^{\boldsymbol{\tau}}(t)\ \mathbf{p}_i^1 = \cdots = \sum_{i=j}^{j} N_{i,0}^{\boldsymbol{\tau}}(t)\ \mathbf{p}_i^0 k = \mathbf{p}_j^0 k$$

with $\mathbf{p}_i^0 = \mathbf{p}_i$ and $\mathbf{p}_i^l = \omega_{i,k-l}^{\boldsymbol{\tau}}(t)\,\mathbf{p}_i^{l-1} + \left(1 - \omega_{i,k-l}^{\boldsymbol{\tau}}(t)\right)\mathbf{p}_{i-1}^{l-1}$



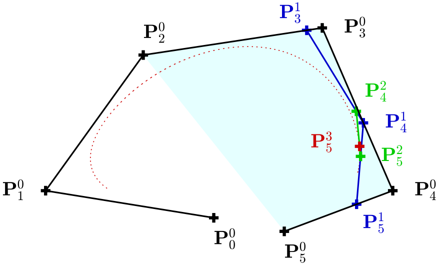
B-splines curves - De Boor - Cox algorithm

De Boor - Cox algorithm

Extension of de Casteljau algorithm to B-splines: $\forall j \in [\![k, n]\!] \ \ \forall t \in [\tau_j, \tau_{j+1}[$

$$\mathcal{B}_{\mathbf{P},\boldsymbol{\tau}}(t) = \sum_{i=j-k}^{j} N^{\boldsymbol{\tau}}_{i,k}(t)\ \mathbf{p}^0_i = \sum_{i=j-k+1}^{j} N^{\boldsymbol{\tau}}_{i,k-1}(t)\ \mathbf{p}^1_i = \cdots = \sum_{i=j}^{j} N^{\boldsymbol{\tau}}_{i,0}(t)\ \mathbf{p}^0_i k = \mathbf{p}^0_j k$$

with $\mathbf{p}^0_i = \mathbf{p}_i$ and $\mathbf{p}^l_i = \omega^{\boldsymbol{\tau}}_{i,k-l}(t)\,\mathbf{p}^{l-1}_i + \left(1 - \omega^{\boldsymbol{\tau}}_{i,k-l}(t)\right)\mathbf{p}^{l-1}_{i-1}$



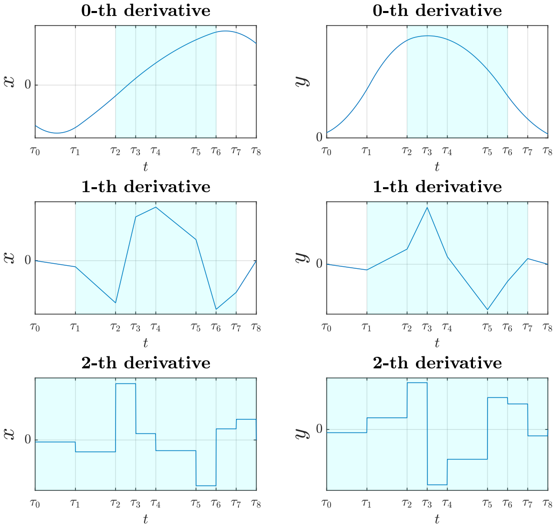
B-splines curves - Derivatives

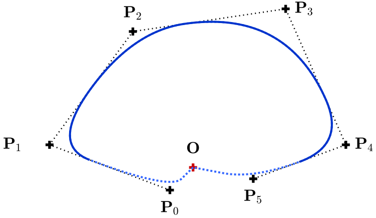
Derivative

The derivative of a B-splines curve of degree $k > 0$ **is a also a B-spline curve**, with the same knots, a degree $k-1$ and $n+1$ control points given by a linear combination of the original ones

$$\mathcal{B}'_{\mathbf{P},\boldsymbol{\tau}} = \mathcal{B}_{\mathbf{P}^{(1)},\boldsymbol{\tau}} \text{ with } \begin{cases} \mathbf{p}_0^{(1)} = \dfrac{k}{\tau_k - \tau_0}\mathbf{p}_0 \\ \forall i \in [\![1, n]\!] \;\; \mathbf{p}_i^{(1)} = \dfrac{k}{\tau_{i+k} - \tau_i}(\mathbf{p}_i - \mathbf{p}_{i-1}) \\ \mathbf{p}_{n+1}^{(1)} = -\dfrac{k}{\tau_{n+k+1} - \tau_{n+1}}\mathbf{p}_n \end{cases} \tag{17}$$

B-splines curves - Derivatives

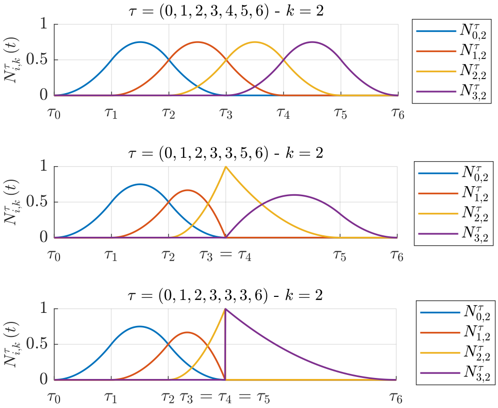




B-splines curves - Knots multiplicity/Continuity

Multiplicity

The number of repetition of a knot τ_i in the knot vector is called its multiplicity $r_i \in \mathbb{N}^*$



Continuity

For $i \in [\![0, m-k-1]\!]$ and $j \in [\![i, i+k]\!]$

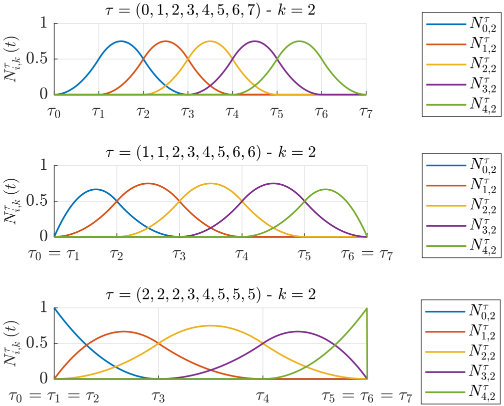
$N^{\tau}_{i,k}$ is $\mathcal{C}^{\infty}$ in $]\tau_j, \tau_{j+1}[$

$N^{\tau}_{i,k}$ is $\mathcal{C}^{k-r_j}$ in a neighborhood of τ_j

$N^{\tau}_{i,k}$ is $\mathcal{C}^{k-r_{j+1}}$ in a neighborhood of τ_{j+1}

B-splines curves - Clamped B-splines

If the first $k+1$ knots and the $k+1$ last knots are equal as well, then the B-spline is "clamped" to its first and last control points



$$\big(\underbrace{\tau_k,\ \dots\ ,\tau_k,}_{k+1 \text{ knots}}\ \underbrace{\tau_{k+1}\ \dots\ ,\ \tau_n,}_{n-k \text{ knots}}\ \underbrace{\tau_{n+1},\ \dots\ ,\ \tau_{n+1}}_{k+1 \text{ knots}}\big)$$

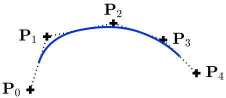
Remark

A B-spline can be converted into a clamped one by **knot insertion** (Boehm's algorithm)

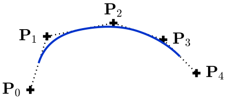
B-splines curves - Clamped B-splines

If the first $k+1$ knots and the $k+1$ last knots are equal as well, then the B-spline is "clamped" to its first and last control points

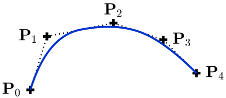
$\tau = (0, 1, 2, 3, 4, 5, 6, 7)$ - $k = 2$



$\tau = (1, 1, 2, 3, 4, 5, 6, 6)$ - $k = 2$



$\tau = (2, 2, 2, 3, 4, 5, 5, 5)$ - $k = 2$



$$\big(\underbrace{\tau_k, \ \dots \ , \tau_k,}_{k+1 \text{ knots}} \ \underbrace{\tau_{k+1} \ \dots \ , \ \tau_n,}_{n-k \text{ knots}} \ \underbrace{\tau_{n+1}, \ \dots \ , \ \tau_{n+1}}_{k+1 \text{ knots}}\big)$$

Remark

A B-spline can be converted into a clamped one by **knot insertion** (Boehm's algorithm)

B-splines curves - A more powerful tool

Share the **same main properties as Bézier curves** (convex hull property and derivatives/integrals are also B-splines) with the **advantages of piecewise polynomials**...

...at the price of **more parameters** (knots and degree) and a **less intuitive definition**

Remark

B-splines can only **approximate** conic sections (parabola, circle/ellipses, hyperbola)
Non Uniform Rational B-Splines (NURBS) allow to parameterize exactly these curves

$$\mathcal{B}_{\mathbf{P},\boldsymbol{\tau},\mathbf{w}}(t) = \frac{\sum_{i=0}^{n} w_i \, N_{i,k}^{\boldsymbol{\tau}}(t) \, \mathbf{p}_i}{\sum_{i=0}^{n} w_i \, N_{i,k}^{\boldsymbol{\tau}}(t)}$$

B-splines curves - References

"Would you like to know more?"

Degree elevation, knot insertion/removal, conversion B-spline curve to Bézier curves...

- Prautzsch, H., Boehm, W., & Paluszny, M., *Bézier and B-spline techniques*, Springer Science & Business Media, 2013
- Piegl, L., & Tiller, W. (2012), *The NURBS book*. Springer Science & Business Media, 2012